

English Abstract of CN 1372769

The patent refers to the field of 'electric digital data processing'. An audio/video (or audiovisual, 'A/V') signal processing apparatus and method for extracting a compact representation of a multimedia description and transcoding hints metadata for transcoding between different (e.g., MPEG) compressed content representations, manipulating (e.g., MPEG compressed) bitstream parameters such as frame rate, bit rate, session size, quantization parameters, and picture coding type structure (e.g., group of pictures, or 'GOP'), classifying A/V content, and retrieving multimedia information.

[19]中华人民共和国国家知识产权局

[51] Int. Cl⁷

H04N 7/26

[12] 发明专利申请公开说明书

G06F 17/30 H04N 7/24

H04N 7/50 H04N 7/36

[21] 申请号 01800759.7

[43] 公开日 2002 年 10 月 2 日

[11] 公开号 CN 1372769A

[22] 申请日 2001.3.13 [21] 申请号 01800759.7

[30] 优先权

[32] 2000.3.13 [33] JP [31] 68720/00

[32] 2000.5.16 [33] US [31] 60/204,729

[86] 国际申请 PCT/JP01/01982 2001.3.13

[87] 国际公布 WO01/69936 英 2001.9.20

[85] 进入国家阶段日期 2001.11.30

[71] 申请人 索尼公司

地址 日本东京都

[72] 发明人 彼得·库恩

[74] 专利代理机构 北京市柳沈律师事务所

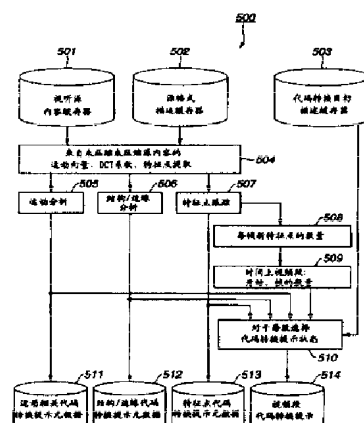
代理人 吕晓章 李晓舒

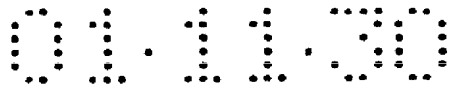
权利要求书 9 页 说明书 17 页 附图页数 13 页

[54] 发明名称 产生压缩的代码转换提示元数据的方法和
和设备

[57] 摘要

一种音频/视频(或视听即“A/V”)信号处理设备和
方法,用于提取多媒体描述的压缩表示和代码转换提示
元数据,用于在不同的(例如,MPEG)压缩内容表示之
间进行代码转换,操作(例如,MPEG 压缩的)位流参数如
帧频、位速率、对话尺寸、量化参数、以及图像编码类型
结构(例如,图像组或“GOP”),将 A/V 内容分类,以及
检索多媒体信息。





权 利 要 求 书

1. 一种用于处理提供的视频/音频信号的视频/音频信号处理方法，包括下面的步骤：

- 5 描述代码转换目标位流参数；
 提取代码转换提示元数据；
 存储代码转换提示元数据；
 将 A/V 材料分为段；
 将代码转换提示元数据与分开的 A/V 段相关；以及
10 代码转换 A/V 材料。

2. 如权利要求 1 所述的视频/音频信号处理方法，其中描述代码转换目标位流参数包括下面的步骤：

- 规定压缩图像第二位流的位速率；
 规定压缩图像第二位流的画面尺寸；
15 规定压缩图像第二位流的每秒的帧数量；
 规定压缩图像第二位流的像素宽高比；
 规定压缩图像第二位流的每个像素的色彩深度；
 规定是否逐行格式用于压缩图像的第二位流；
 规定是否隔行格式用于压缩图像的第二位流；
20 规定是否将帧画面用于压缩图像的第二位流；
 规定是否将保留画面用于压缩图像的第二位流；以及
 规定压缩图像第二位流的压缩方法。

3. 如权利要求 2 所述的视频/音频信号处理方法，其中描述代码转换目标位流参数的步骤还包括规定使用如 MPEG(活动图像专家组)规定的压缩标准
25 的步骤。

4. 如权利要求 2 所述的视频/音频信号处理方法，其中描述代码转换目标位流参数的步骤还包括规定使用如 ITU-T(国际电信联盟技术标准组)规定的压缩标准的步骤。

5. 如权利要求 1 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤包括：
30 接收具有第一 GOP 结构的压缩图像数据的第一位流；

从第一位流获得第一运动信息；
获得第一段的结构/边缘信息；
从第一位流获得特征点和相关的运动信息；以及
从第一位流获得感兴趣信息的区域。

5 6. 如权利要求 5 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤还包括将第一运动信息存储为代码转换提示元数据。

7. 如权利要求 5 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤还包括将运动相关代码转换提示元数据表示为参数运动模型的参数的步骤。

10 8. 如权利要求 7 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤还包括使用参数运动模型来描述随后的矩形视频帧内全局运动的步骤。

15 9. 如权利要求 7 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤还包括使用参数运动模型来描述任意形状规定区域内运动的步骤。

10. 如权利要求 9 所述的视频/音频信号处理方法，其中使用参数运动模型以描述如 MPEG-4 内使用的任意形状的规定区域内的运动。

20 11. 如权利要求 5 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤还包括将运动相关代码转换提示元数据表示为包含在压缩图像数据第一位流内的运动向量阵列的步骤。

12. 如权利要求 5 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤还包括将运动相关代码转换提示元数据表示为从包含在压缩图像数据第一位流的运动向量获得的运动向量阵列的步骤。

25 13. 如权利要求 5 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤还包括将运动相关代码转换提示元数据表示为具有相关运动向量的特征点列表，它们在随后的帧内被跟踪。

14. 如权利要求 5 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤还包括将运动相关代码转换提示元数据表示为具有相关运动向量的特征点列表，它们在随后的帧内在任意形状区域内被跟踪。

30 15. 如权利要求 5 所述的视频/音频信号处理方法，其中提取代码转换提示元数据的步骤还包括将结构相关的代码转换提示元数据表示为 DCT 系数

列表中的一个和从那里获得的一个测量值(平均值、最小值、最大值、变量、标准偏差中的一个)。

16. 如权利要求 5 所述的视频/音频信号处理方法, 其中提取代码转换提示元数据的步骤还包括将边缘相关的代码转换提示元数据表示为 DCT 系数列表中的一个和从那里获得的一个测量值(平均值、最小值、最大值、变量、标准偏差中的一个)。

17. 如权利要求 5 所述的视频/音频信号处理方法, 其中提取代码转换提示元数据的步骤还包括将特征点和有关的运动相关的代码转换提示元数据表示为列表。

10 18. 如权利要求 5 所述的视频/音频信号处理方法, 其中提取代码转换提示元数据的步骤还包括将编码复杂性相关的代码转换提示元数据表示为从特征点使用期限列表获得的复杂性度量, 这些特征点通过使用从一个帧到下一个帧的许多丢失的和新的特征点在随后的帧内被跟踪。

15 19. 如权利要求 1 所述的视频/音频信号处理方法, 其中存储代码转换提示元数据的步骤包括保留一个缓存器的步骤, 该缓存器包含几种情况下的代码转换提示元数据。

20. 如权利要求 19 所述的视频/音频信号处理方法, 其中存储代码转换提示元数据的步骤还包括存储各个通用的代码转换提示元数据用于几个目标装置的步骤。

20 21. 如权利要求 19 所述的视频/音频信号处理方法, 其中存储代码转换提示元数据的步骤还包括存储通用的代码转换提示元数据用于不同场景活动的 A/V 段的步骤。

22. 如权利要求 1 所述的视频/音频信号处理方法, 其中将 A/V 材料分为段的步骤包括:

25 使用具有相关运动向量的特征点;
跟踪特征点并且保持特征点的使用期限; 以及
根据不能逐帧跟踪的特征点的数量, 确定用于代码转换的新 A/V 段。

23. 如权利要求 1 所述的视频/音频信号处理方法, 其中将代码转换提示元数据与分开的 A/V 段相关的步骤包括:

30 计算每帧新特征点的数量;
确定是否新特征点的数量超过某些门限值; 以及根据一些代码转换提示

状态中所述确定的一个来选择。

24. 如权利要求 1 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤包括:

接收具有第一 GOP 结构的压缩图像数据的第一位流;

5 从第一位流中提取代码转换提示元数据;

利用与第一位流相关的代码转换提示元数据以便于代码转换; 以及
输出第二位流。

25. 如权利要求 24 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括利用与第一位流的时间段相关的代码转换提示元数据以便
10 于代码转换。

26. 如权利要求 24 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括利用与第一位流的空间段相关的代码转换提示元数据以便于代码转换。

27. 如权利要求 24 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括利用包含在代码转换提示元数据中的运动信息来外插第二
15 运动信息, 用于具有与第一 GOP 结构不同的第二 GOP 结构的压缩图像数据的第二位流。

28. 如权利要求 24 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括控制第二位流的位速率的步骤, 使得第一位流的位速率不
20 同于第二位流的位速率。

29. 如权利要求 28 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括调节第一位流表示的图像尺寸的步骤, 使得第二位流表示
的图像的尺寸不同于第一位流表示的图像的尺寸。

30. 如权利要求 24 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括调节第一位流表示的图像尺寸的步骤, 使得第二位流表示
25 的图像的尺寸不同于第一位流表示的图像的尺寸。

31. 如权利要求 30 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括当第一位流表示的图像被编码为帧图像时, 将第二位流表示
的图像编码为字段图像的步骤。

32. 如权利要求 30 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括当第一位流表示的图像被编码为字段图像时, 将第二位流

表示的图像编码为帧图像的步骤。

33. 如权利要求 30 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括当第一位流表示的图像作为逐行序列接收时, 将第一位流表示的图像进行隔行处理的步骤, 使得第二位流表示的图像作为隔行序列输出。

34. 如权利要求 30 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括当第一位流表示的图像作为隔行序列接收时, 将第一位流表示的图像去隔行处理的步骤, 使得第二位流表示的图像作为逐行序列输出。

35. 如权利要求 24 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括当第一位流表示的图像被编码为帧图像时, 将第二位流表示的图像编码为字段图像的步骤。

36. 如权利要求 24 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括当第一位流表示的图像被编码为字段图像时, 将第二位流表示的图像编码为帧图像的步骤。

37. 如权利要求 24 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括当第一位流表示的图像作为逐行序列接收时, 将第一位流表示的图像进行隔行处理的步骤, 使得第二位流表示的图像作为隔行序列输出。

38. 如权利要求 24 所述的视频/音频信号处理方法, 其中代码转换 A/V 材料的步骤还包括当第一位流表示的图像作为隔行序列接收时, 将第一位流表示的图像去隔行处理的步骤, 使得第二位流表示的图像作为逐行序列输出。

39. 一种代码转换方法, 包括下面的步骤:

接收表示第一尺寸画面的压缩图像数据的第一位流;
从第一位流中提取第一运动相关的代码转换提示元数据;
存储第一个运动相关的代码转换提示元数据;
利用存储的第一运动相关的代码转换提示元数据来外插第二运动信息, 用于表示不同于第一尺寸的第二尺寸的图像的压缩图像数据的第二位流; 及
输出第二位流。

40. 一种代码转换方法, 包括下面的步骤:

接收表示规定隔行序列的画面的压缩图像数据的第一位流；
从第一位流中提取第一运动相关的代码转换提示元数据；
存储第一运动相关的代码转换提示元数据；
利用存储的第一个运动相关的代码转换提示元数据来外插第二运动信

5 息，用于表示规定逐行序列的图像的压缩图像数据的第二位流；以及
输出第二位流。

41. 一种代码转换方法，包括下面的步骤：

接收表示规定逐行序列的画面的压缩图像数据的第一位流；
从第一位流中提取第一运动相关的代码转换提示元数据；
10 存储第一运动相关的代码转换提示元数据；
利用存储的第一运动相关的代码转换提示元数据来外插第二运动信息，

用于表示规定隔行序列的图像的压缩图像数据的第二位流；以及
输出第二位流。

42. 一种代码转换方法，包括下面的步骤：

15 接收表示帧画面的压缩图像数据的第一位流；
从第一位流中提取第一运动相关的代码转换提示元数据；
存储第一运动相关的代码转换提示元数据；
利用存储的第一运动相关的代码转换提示元数据来外插第二运动信息，

用于表示字段图像的压缩图像数据的第二位流；以及
20 输出第二位流。

43. 一种代码转换方法，包括下面的步骤：

接收表示字段画面的压缩图像数据的第一位流；
从第一位流中提取第一运动相关的代码转换提示元数据；
存储第一运动相关的代码转换提示元数据；
25 利用存储的第一运动相关的代码转换提示元数据来外插第二运动信息，

用于表示帧图像的压缩图像数据的第二位流；以及
输出第二位流。

44. 一种代码转换方法，包括下面的步骤：

接收表示主要图像的压缩图像数据的第一位流；
30 从第一位流中提取第一运动相关的代码转换提示元数据；
存储第一运动相关的代码转换提示元数据；

利用存储的第一运动相关的代码转换提示元数据来外插第二运动信息，
用于表示一部分主要图像的压缩图像数据的第二位流；以及
输出第二位流。

45. 一种代码转换方法，包括下面的步骤：

- 5 接收具有多个编码参数的压缩图像数据的第一位流，这些参数包括 GOP
结构、图像尺寸、位速率、帧画面格式、字段画面格式、逐行序列、以及隔
行序列中至少一个；

从第一位流中提取第一运动相关的代码转换提示元数据；

存储第一运动相关的代码转换提示元数据；

- 10 利用存储的第一运动相关的代码转换提示元数据来外插第二运动信息，
用于具有若干个编码参数的压缩图像数据的第二位流，使得第二位流的编码
参数中一个或多个与第一位流的编码参数不同；以及

输出第二位流。

46. 一种代码转换方法，包括下面的步骤：

- 15 接收表示第一尺寸的画面的压缩图像数据的第一位流；

从第一位流中提取第一特征点运动相关的代码转换提示元数据；

存储第一特征点运动相关的代码转换提示元数据；

- 20 利用存储的第一特征点运动相关的代码转换提示元数据来外插第二运动
信息，用于表示不同于第一尺寸的第二尺寸的画面的压缩图像数据的第二位
流；以及

输出第二位流。

47. 一种代码转换方法，包括下面的步骤：

接收表示规定隔行序列的画面的压缩图像数据的第一位流；

从第一位流中提取第一特征点运动相关的代码转换提示元数据；

- 25 存储第一特征点运动相关的代码转换提示元数据；

利用存储的第一特征点运动相关的代码转换提示元数据来外插第二运动
信息，用于表示规定逐行序列的画面的压缩图像数据的第二位流；以及
输出第二位流。

48. 一种代码转换方法，包括下面的步骤：

- 30 接收表示规定逐行序列的画面的压缩图像数据的第一位流；

从第一位流中提取第一特征点运动相关的代码转换提示元数据；

存储第一特征点运动相关的代码转换提示元数据；

利用存储的第一特征点运动相关的代码转换提示元数据来外插第二运动信息，用于表示规定隔行序列的画面的压缩图像数据的第二位流；以及输出第二位流。

5 49. 一种代码转换方法，包括下面的步骤：

接收表示帧画面的压缩图像数据的第一位流；

从第一位流中提取第一特征点运动相关的代码转换提示元数据；

存储第一特征点运动相关的代码转换提示元数据；

10 利用存储的第一特征点运动相关的代码转换提示元数据来外插第二运动信息，用于表示字段画面的压缩图像数据的第二位流；以及输出第二位流。

50. 一种代码转换方法，包括下面的步骤：

接收表示字段画面的压缩图像数据的第一位流；

从第一位流中提取第一特征点运动相关的代码转换提示元数据；

15 存储第一特征点运动相关的代码转换提示元数据；

利用存储的第一特征点运动相关的代码转换提示元数据来外插第二运动信息，用于表示帧画面的压缩图像数据的第二位流；以及输出第二位流。

51. 一种代码转换方法，包括下面的步骤：

20 接收表示主要图像的压缩图像数据的第一位流；

从第一位流中提取第一特征点运动相关的代码转换提示元数据；

存储第一特征点运动相关的代码转换提示元数据；

25 利用存储的第一特征点运动相关的代码转换提示元数据来外插第二运动信息，用于表示一部分主要图像的压缩图像数据的第二位流；以及输出第二位流。

52. 一种代码转换方法，包括下面的步骤：

接收具有多个编码参数的压缩图像数据的第一位流，这些参数包括 GOP 结构、画面尺寸、位速率、帧画面格式、字段画面格式、逐行序列、以及隔行序列中至少一个；

30 从第一位流中提取第一特征点运动相关的代码转换提示元数据；

存储第一特征点运动相关的代码转换提示元数据；

利用存储的第一特征点运动相关的代码转换提示元数据来外插第二运动信息，用于具有多个编码参数的压缩图像数据的第二位流，使得第二位流的编码参数的一个或多个与第一位流的编码参数不同；以及

输出第二位流。

- 5 53. 一种用于处理提供的视频信号的视频处理方法，包括下面的步骤：
接收源视频；以及

使用运动元数据、结构/边缘元数据、以及特征点和相关的运动元数据中的一个，将源视频的内容分类，包括每帧许多新特征点。

- 10 54. 如权利要求 53 所述的视频处理方法，其中所述方法用于确定代码转换的代码转换参数设置。

55. 如权利要求 53 所述的视频处理方法，其中所述方法用于根据源视频的内容分类来组织视听材料。

56. 一种用于处理提供的视频/音频信号的设备，包括：

- 15 用于存储代码转换目标位流参数中至少一个描述的目标缓存器；
根据这至少一个描述，用于提取代码转换提示元数据的提取单元；
用于存储代码转换提示元数据的缓存器；
用于将 A/V 材料分为段的分段单元；以及
用于将代码转换提示元数据与分开的 A/V 段相关并且代码转换 A/V 材料的代码转换单元。

- 20 57. 一种代码转换设备，包括：

用于接收表示第一尺寸画面的压缩图像数据的第一位流的输入端；

用于从第一位流中提取代码转换提示元数据的代码转换提示元数据提取单元；

用于存储代码转换提示元数据的缓存器；

- 25 处理单元，利用存储的代码转换提示元数据来外插运动信息，用于与第一位流不同的压缩图像数据的第二位流；以及

输出第二位流的输出端。

58. 一种用于处理提供的视频信号的设备，包括：

用于接收源视频的输入端；以及

- 30 处理器，利用运动元数据、结构/边缘元数据、以及特征点和相关的运动元数据中的一个，将源视频的内容分类，包括每帧许多新特征点。

说明书

产生压缩的代码转换提示元数据的方法和设备

5 技术领域

本发明涉及音频/视频(或视听“A/V”)信号处理方法和 A/V 信号处理设备,用于提取多媒体描述的压缩表示和不同(例如 MPEG)压缩内容表示之间进行代码转换的代码转换提示元数据,操作(例如 MPEG 压缩的)位流参数如帧频、位速率、对话大小、量化参数以及图像编码类型结构如图像组或“GOP”,分类 A/V 内容,以及检索多媒体信息。

背景技术

A/V 内容越来越多地经过光纤、无线和有线网络来传送。因为这些网络由不同的网络带宽约束来表征,所以需要由导致不同主观视觉质量的不同位速率来表示 A/V 内容。对于 A/V 内容压缩表示的其他要求通过 A/V 终端的屏幕大小、计算性能和存储器约束来产生。

因此,以如活动图像专家组(“MPEG”)规定的压缩格式存储的 A/V 内容必须转换为例如不同的位速率、帧频、屏幕大小,并且适应不同的译码复杂性和不同 A/V 终端的存储约束。

为了避免对不同的网络带宽和不同的 A/V 终端存储相同 A/V 内容的多个压缩的表示,以压缩的 MPEG 格式存储的 A/V 内容可以被代码转换为不同的 MPEG 格式。

对于视频代码转换,可以参照下面的内容:

W009838800A1: O.H.Werner, N.D.Wells, M.J.Knee: 采用改进量化的数字压缩编码, 1999, 提出了一种自适应量化方案;

US5870146: Zhu; Qin-Fan: 用于数字视频代码转换的装置和方法, 1999;

W009929113A1: Nilsson, Michael, Erling; Ghanbari, Mohammed: 代码转换, 1999;

US5805224: Keesman; Gerrit J, Van Otterloo; Petrus J.: 用于代码转换视频信号的方法和装置, 1998;

W009943162ALGolin, Stuart, Jay: 用于代码转换视频序列的运动向量

外插，1999；

US5838664：Polomski；Mark D.：采用数字代码转换的视频电话会议系统，1998；

W009957673A2：Balliol，Nicolas：数据流的代码转换，1999；

- 5 US5808570：Bakhmutsky；Michael：用于双匹配霍夫曼代码转换的装置和方法以及具有利用它们的两个码字位流分段的高性能变量长度译码器，1998；

W009905870A2：Lemaguet，Yann：在视频序列和相应装置之间转换的方法，1999；以及

- 10 W009923560A1：LUDWIG，Lester；BROWN，William；YUL，Inn，J.；VUONG，Anh，T.，VANDERLIPPE，Richard；BURNETT，Gerald；LAUWERS，Chris；LUI，Richard；APPLEBAUM，Daniel：可伸缩的网络多媒体系统和应用，1999。

- 15 然而，在视频代码转换方面，这些专利中没有一个揭示或建议使用代码转换提示元数据信息以便于 A/V 代码转换。

- 20 电影电视工程师协会(“SMPTE”)推荐了一种用于 MPEG-2 视频记录数据集的电视标准(327M-2000)，它对于源格式的每个宏块提供使用 256 位的重新编码的元数据。但是，这种代码转换提示元数据的提取和表示具有一些缺点。例如，根据推荐的标准，对于 A/V 源内容的每单个帧和宏块提取代码转换提示元数据(如 GOP 结构、量化器设置、运动向量等)。这种方法的优点是提供详细的和内容自适应的代码转换提示，便于代码转换，同时大大地保留了主观 A/V 质量。然而，代码转换提示元数据的尺寸是非常大的。在推荐标准的一种特定实施方式中，MPEG 视频的每个宏块存储 256 位代码转换提示元数据。这样大的代码转换提示元数据量不便于到本地(例如住家)A/V 内容服务器的广播分配。因此。关于代码转换提示元数据的推荐标准只限于广播制作室应用。
- 25

- 30 用于代码转换提示元数据提取和表示的另一种技术包括收集通用的代码转换提示元数据，用于以特定的位速率将压缩的 A/V 源内容代码转换到另一个压缩格式和位速率。然而，这种技术的缺点是不考虑代码转换内容的特有特性。例如，在源内容中，A/V 特性可以从具有有限运动量和很少细节(例如新闻锚固场景)的 A/V 段改变为描述快速运动和许多细节(例如运动事件场

景)的另一个 A/V 段。根据这种技术, 可能选择错误引导代码转换提示元数据, 它不适用于表示两种视频段不同的特性, 因此, 导致不好的 A/V 质量和错误的位速率分配。

5 本发明公开

根据前面所述, 本发明的目的是提供一种用于提取压缩和 A/V 内容自适应多媒体描述和代码转换提示元数据表示的方法和设备。

10 本发明的另一个目的是提供一种代码转换方法和设备, 代码转换方法的一个要求是允许实时执行而没有明显的延迟, 并且抑制计算复杂性。对于代码转换方法的第二个要求是尽可能多地保留主观 A/V 质量。为了便于代码转换方法对于不同的压缩目标格式达到这两个要求, 可以提前产生并且分开存储或与压缩的 A/V 内容一起存储代码转换提示元数据。本发明的另一个目的是提供一种高度压缩的表示, 以减少存储尺寸并且便于多媒体描述和代码转换提示元数据的分配(例如, 到本地 A/V 内容服务器的广播)。

15 因此, 本发明的目的是为代码转换系统提供: 1)经过代码转换处理保持 A/V 质量, 以及 2)限制计算复杂性以便使得实时应用以最小的延迟进行。根据本发明的实施例, 包括代码转换提示的附加的数据(元数据)可以与压缩的 A/V 内容相关。

20 通过说明书和附图的描述, 本发明的其它目的和优点将会变得更加明显。

本发明的设备和方法提供自动代码转换提示元数据的提取和压缩表示。

25 本发明的领域是通过使用支持的代码转换元数据, 将一种压缩格式的压缩的 A/V 内容代码转换为另一种格式的 A/V 内容。术语“代码转换”包括但不限于改变压缩的格式(例如, 从 MPEG-2 格式转换为 MPEG-4 格式)、帧频转换、位速率转换、对话尺寸转换、屏幕尺寸转换、画面编码类型转换等。

本发明也可以应用于自动的视频分类, 使用前面所述的代码转换提示状态作为视频的不同场景活动的类别。

30 因此, 本发明包括若干个步骤并且这些步骤的一个或多个相关于其它步骤中的每一个, 以及体现结构特征的设备, 适用于实现这些步骤的元件组合和部件配置, 所有这些都作为下面公开的详细内容进行示范, 本发明的范围将在权利要求书中表示。

代码转换单元 106、代码转换提示元数据提取单元 104、以及 A/V 代码转换提示元数据存储缓冲器 105。A/V 内容存储器 103 存储来自不同源的压缩的 A/V 材料，这些源具有不同的位速率和不同的主观质量。例如，A/V 内容存储器 103 可以包含来自便携式数字视频(“DV”)摄像机 111 的家庭视频，来自 MPEG-4 互联网络摄像机 112 的具有非常低位速率(比如说 10 kbit/s)的 MPEG-4 压缩视频，以及来自广播服务 101 的大约 5 Mbit/s 的 MPEG-2 Main Profile at Main Level(“MP@ML”)压缩广播视频，在某些情况下它已经与代码转换提示元数据相关联。A/V 内容服务器 102 也可以包含以相当高位速率的高清晰度压缩 MPEG 视频。

如图 1 所示，A/V 内容服务器 102 连接到网络 113，该网络可以是有线或无线家网。一些具有不同特性的 A/V 终端也可以连接到网络 113，包括但不限于：无线 MPEG-4 A/V 个人数字助理(“PDA”)107、用于高清晰度电视娱乐的高分辨率 A/V 终端 108、A/V 游戏控制台 109、以及基于国际电信联盟技术标准组(“ITU-T”)的可视电话 110。A/V 终端 107、108、109 和 110 可以采用不同的位速率传输性能(取决于有线电视或无线电链路)连接到家网 113。

另外，根据计算功率、存储器、屏幕尺寸、视频帧频、以及网络位速率可以限制无线视频 PDA 107。因此，A/V 代码转换单元 106 可以将例如欧洲 5 Mbit/s MPEG-2 广播电视的每秒钟 25 帧(“fps”)和包含在 A/V 内容服务器 102 中的 720×480 像素代码转换为 MPEG-4 500 kbit/s 15fps 视频用于无线传输，并且通过无线 MPEG-4 视频 PDA 107 显示在 352×240 像素显示器上。A/V 代码转换单元 106 可以使用来自缓存器 105 的代码转换提示元数据，实时地将 A/V 内容的压缩的源位速率代码转换为每个特定的目标 A/V 终端 107、108、109 和 110 的性能。代码转换提示元数据在代码转换提示元数据提取单元 104 中产生，或者它们可以由广播服务程序 101 分配。

如图 1 所示，采用源格式的压缩位流(以后称为“第一位流”)116 从 A/V 内容缓存器 103 传送到 A/V 代码转换单元 106。采用目标格式的位流(以后称为“第二位流”)115 在代码转换单元 106 的代码转换之后传送到家网 113。来自家网 113 的例如采用压缩 DV 格式的内容经过链路 114 存储在 A/V 内容存储器 103 中。

图 2 根据本发明实施例，说明代码转换提示提取、代码转换提示存储以

及代码转换处理。如图 2 所示，缓存器 201 包含采用源格式的 A/V 内容。缓存器 202 包含源格式的描述，如位速率、压缩方法、GOP 结构、屏幕尺寸、隔行或逐行的格式等。缓存器 203 包含目标格式的描述，如位速率、压缩方法、GOP 结构、屏幕尺寸、隔行或逐行的格式等。代码转换提示提取单元 207 从 A/V 缓存器 201 读出采用压缩源格式的 A/V 内容，以及来自缓存器 202 的源格式描述和来自缓存器 203 的代码转换目标格式描述。在代码转换提示提取单元 207 计算代码转换提示之后，代码转换提示存储在代码转换提示元数据缓存器 206 中。A/V 代码转换单元 205 从 A/V 内容缓存器 201 读出采用源格式的第一位流 204，并且借助于存储在缓存器 206 的代码转换提示元数据将源格式转换为目标格式。A/V 代码转换单元 205 输出采用新的压缩目标格式的第二位流 208 到 A/V 目标格式缓存器 209 用于存储。

图 3 和图 4 根据本发明实施例，说明代码转换提示元数据组织的原理。基于 MPEG 的视频压缩使用可预测的方法，这里相邻帧之间的变化被编码。从一个帧到下一个帧具有许多变化的视频内容与帧之间具有较小变化的视频内容相比，要求(为了保持主观质量同时限制位速率)不同的重新编码参数设置。因此，重要的是事先决定重新编码参数。代码转换提示元数据选择主要取决于不可预测的视觉内容的量和特性。新的视觉内容不能由先前帧预测并且可以是使用 DCT 系数密集编码的位速率。这样，本发明方法使用新特征点数量来确定每帧新的内容量，该新特征点的数量不是从先前帧到当前帧跟踪的。

图 3 根据视频的帧数量，描述了每帧新特征点数量的图形(水平轴，时间轴)。部分 301 是一部分视频段，这里在连续帧之间只出现了非常少量的新内容，因此可以选择相应的代码转换提示元数据(例如，大 GOP 尺寸，低帧频，低位速率)。部分 302 包括每帧稍微高一些的新特征点的数量，这意味着选择了描述代码转换提示元数据的状态，它提供了这种情况下最佳的代码转换参数(例如，稍微小一些 GOP 尺寸，较高位速率)。部分 303 描述了每帧具有较大的新特征点数量的代码转换元数据提示状态，因此，每个场景具有较高的新内容量。这样，选择了较小 M 值(I/P 帧距离)和较高位速率。

图 4 描述了包括三个离散的代码转换提示元数据状态的代码转换提示元数据状态图的基本组织的例子。每个离散的代码转换状态可以包含用于 GOP 结构的元数据、量化器参数、位速率、屏幕尺寸等。这些代码转换提示参数

可以具有一个固定值或可以是另一个参数的函数。例如，GOP 长度可以是每帧新特征点数量的离散函数，量化器参数可以是从小 DCT 系数获得的边缘和结构活动性的函数。在这个例子中可以选择三个代码转换提示元数据状态的每一个来适应三种不同的编码情况。如图 4 所示，状态“3”403 被选择用于每帧高运动量和低新内容量，并且表示对这种内容的代码转换提示元数据的最佳状态。状态“2”402 被选择用于具有高边缘活动性的低运动量和高内容量，它可能需要使用许多位。例如，状态“1”401 被选择适用于具有低场景活动性的 A/V 内容的代码转换处理。还提供其它特别的代码转换提示元数据状态用于视频编辑效果，如不同的交叉衰落效果、突然的场景变化、或者两个场景之间的黑图像。视频编辑效果的位置可以手工地、半自动地或全自动地检测。

图 5 根据本发明实施例，说明了从压缩和未压缩的源内容提取的代码转换提示元数据。如图 5 所示，系统 500 包括 A/V 源内容缓存器 501、源格式描述缓存器 502、以及目标格式描述缓存器 503。

存储器 504 用于存储运动向量、DCT 系数以及来自压缩和未压缩区域的特征点提取。在压缩区域中，来自 P 和 B 宏块的运动向量可以直接从位流中提取。然而，宏块内不存在运动向量。因此，对于 B 和 P 宏块获得的运动向量可以内插用于 I 宏块(见 Roy Wang, Thomas Huang: “在 MPEG 区域的快速摄像运动分析”，IEEE 图像处理国际会议(International Conference on Image Processing), ICIP 99, Kobe, Japan, Oct 1999)。用于宏块内的块的 DCT 系数可以直接从位流中提取。对于 P 和 B 宏块，限定数量的 DCT 系数(DC 和 2 个 AC 系数)可以通过下面描述的方法来获得，即 Shih-Fu Chang, David G.Messerschmid: “MC-DCT 压缩视频的操作和合成”，IEEE 选择通信区域杂志(Journal on Selected Areas in Communications), vol.8, 1996。在 Peter Kuhn 1999 年 12 月的 PCT 专利“用于压缩区域特征点登记和运动估计的方法和装置”中，公开了压缩区域特征点提取和运动估计的示范方法，它在此作为参考。在某些情况下，A/V 源内容可以仅仅以未压缩的格式或以不基于 DCT 和运动补偿原理的压缩格式获得，它由 MPEG-1、MPEG-2、MPEG-4、ITU-T H.261 和 ITU-T H.263 使用。对于 DV 格式，可能只有 DCT 系数可以获得。在这些情况下，运动向量可以通过运动估计方法来获得，参照例如 Peter Kuhn “用于 MPEG-4 运动估计的算法、复杂性分析和 VLSI 体系结构”，Kluwer

Academic Publishers, 1999。DCT 系数可以通过完成基于块的 DCT 变换来获得, 参照 K.R.Rao, P.Yip: “离散余弦变换-算法, 优点、应用”, 学院出版社(Academic Press)1990。在像素区域(未压缩区域)的特征点可以通过例如下面描述的方法获得, 即 Bruce D.Lucas, Takeo Kanade “应用于立体视觉的迭代登记技术”, 人工智能国际联合会议(International Joint Conference on Artificial Intelligence), pp.674-679, 1981。

运动分析部分 505 从存储器 504 的运动向量表示中提取参数运动模型参数。参数运动模型可以具有 6 个和 8 个参数, 参数运动估计可以通过下面描述的方法获得, 即 M.Tekalp: “数字视频处理”, Prentice Hall, 1995。使用运动表示的目的是去除由于延迟和速度原因在代码转换器中的运动估计。因此, 来自源位流的运动输入表示可以用于获得输出表示(目标位流)。例如, 对屏幕尺寸重定尺寸、隔行-逐行转换等可以主要地取决于运动表示。运动表示的参数也可以根据 GOP 结构进行编码判定。结构/边缘分析部件 506 可以基于从位流提取的 DCT 系数, 可以参照例如 K.R.Rao, P Yip “离散余弦变换-算法、优点、应用”, 学院出版社(Academic Press) 1990, 或者 K.W.Chun, K.W. Lim, H.D.Cho, J.B.Ra “用于视频编码的自适应感觉量化算法”, IEEE 消费电子文集(Transactions on Consumer Electronics), Vol.39, No.3, August 1993。

用于压缩区域的特征点跟踪部件 507 可以使用 Peter Kuhn 描述的技术, 即 “用于压缩区域特征点登记和运动估计的方法和设备”, PCT 专利, 1999 年 12 月, 它在此作为参考。处理器 510 计算每帧新特征点的数量。处理器 509 计算时间的视频段, 处理器 510 计算每段的代码转换提示状态。下面参照附图 6、图 7 和图 8 详细地描述根据本发明实施例用于这些计算的方法。

存储器 511 包含运动相关的代码转换提示元数据。存储器 512 包含结构/边缘相关的代码转换提示元数据, 而存储器 513 包含特征点代码转换提示元数据, 下面参照图 15 对它们进行详细描述。存储器 514 包含视频段代码转换提示选择元数据, 参照图 16 来描述它们。现在描述代码转换提示元数据的自动提取、压缩表示以及用途。

图 6 根据本发明实施例, 公开了视频段和代码转换提示状态选择处理。在步骤 601, 初始化一些变量。变量“帧”是源位流的当前帧数量, “n 帧(nframes)”是新视频段(或 GOP, 即图像组)内帧的数量。其它的变量仅用于

这个子程序。在步骤 602, GOP 内帧的数量加 1。在步骤 603, 确定是否新的段/GOP 在帧内开始, 参照图 7 将讨论它的详细内容。如果是这样(“是”), 控制进入到步骤 604, 否则, 进入步骤 615。在步骤 604, 变量“last_gop_start”采用“new_gop_start”值初始化。在步骤 608 和 609, 如果变量“帧”大于 1 则变量“last_gop_stop”设置为“frame-1”。否则, 在步骤 610, “last_gop_stop”设置为 1。接着, 将在图 8 中详细描述的步骤 611, 根据运动参数 605、结构/边缘参数 606、以及特征点数据 607 确定代码转换提示状态。在步骤 612, 代码转换提示元数据输出到代码转换提示元数据缓存器。根据本发明优选实施例, 代码转换提示元数据包括“n 帧”(GOP 内帧的数量), 具有所有参数的代码转换提示状态, 以及新 GOP 的开始帧数量(“new_gop_start”)。在这以后, 变量“n 帧”被设置为 0, 并且当前帧数量“帧”被给予变量“new_gop_start”。然后, 在步骤 615, 检测以确定是否源位流的所有帧已经被处理。如果不是(“否”), 控制转到步骤 614, 这里帧数量被加 1 并且从步骤 602 开始重复该处理。否则, 该处理终止。

图 7 根据本发明实施例, 说明确定新视频段或 GOP 的开始帧和结束帧的方法。在步骤 701, 确定是否来自图 6 的变量“n 帧(nframes)”是 M(它是 I/P 帧距离)的整数倍。如果是这样, 则选择“否”并且在步骤 702, 确定是否当前帧号码是第一帧。如果是这样(“否”), 控制转到步骤 703, 这里确定是否“n 帧”大于 GOP 内帧“gop_min”的最小数。在步骤 702 的结果是“是”的情况下, 新 GOP 在步骤 705 开始。在步骤 703 的结果是“是”的情况下, 新 GOP 在步骤 705 开始。在步骤 703 的结果是“否”的情况下, 控制转到步骤 704, 这里确定是否“n 帧”大于 GOP 内帧“gop_max”的最大数。在步骤 704 的结果是“是”的情况下, GOP 在步骤 706 关闭, 否则该处理终止。

图 8 根据本发明实施例, 说明选择代码转换提示状态用于特定的 GOP 或 A/V 段的处理, 仅考虑每帧新特征点的数量。根据说明的基本构思, 可以使用来自参数运动估计的上述的运动参数以及从 DCT 系数获得的结构/边缘参数来实现类似的判定结构。应该注意描述的类别和算法也可用于根据运动、边缘活动性、每帧新内容等将 A/V 材料分类, 使得有更高级别的 A/V 分类。在这种情况下, 代码转换提示状态将表示不同内容材料的特定分类。现在参照图 8, 在步骤 801, 变量“frame_no”、“last_gop_start”、“sum”和

“new_seg”被初始化。变量“frame no”被给予“last_gop_start”参数的内容，变量“sum”和“new_seg”被初始化为零。然后，在步骤 802，变量“sum”的内容加上当前帧的新特征点的数量(“frame_no”)。在步骤 803，确定是否变量“frame_no”小于变量“last_gop_stop”。如果是这样(“是”)，重复
 5 步骤 802，否则，控制转到步骤 804。在步骤 804，确定是否变量“sum”的值小于预定参数“summax”的八分之一。参数“summax”是一个常数，它表示可以逐帧跟踪的最大特征点数量乘以帧“last_gop_start”和“last_gop_stop”之间的帧数量。本发明实施例中它可以具有值 200。如果在步骤 804 的结果是“是”，在步骤 806 选择代码转换提示状态 1，它用于
 10 图 8 的表 1 所示的参数。否则，在步骤 805，确定是否变量“sum”的值小于预定参数“summax”的四分之一。如果是这样(“是”)，则为代码转换提示状态 2，如步骤 807 选择的表 1 所示。如果不是(“否”)，则在步骤 808 选择代码转换提示状态 3(如表 1 所示)并且该处理终止。应该注意步骤 804 和 805 的判定门限值取决于代码转换提示状态的规定和数量。

15 代码转换提示元数据描述

为了解释元数据，可以使用伪 C 编码形式。可以使用如 MPEG-7 元数据标准中规定的描述的缩写 D 和描述方案的缩写 DS。

图 9 根据本发明实施例，描述了通用的 A/V DS 901 内代码转换提示元数据的结构组织。如图 9 所示，段 DS 904 和媒体信息 DS 902 从通用的 A/V DS
 20 901 中获得。段分解 906 从段 DS 904 中获得，视频段 DS 907 和移动区域 DS 907 从段分解 906 中获得。基于段的代码转换提示 DS 909 从视频段 DS 907 中获得，将参照图 14 详细描述 909。视频段 DS 907 存取一个或多个代码转换提示状态 DS 911，它参照图 16 被详细地描述。参照图 14 详细描述的基于段的代码转换提示 DS 910 由移动区域 DS 908 获得用于移动区域，它存取
 25 一个或多个代码转换提示状态 DS 912，它参照图 16 被详细地描述。从媒体信息 DS 902 获得媒体分布(Media Profile) DS 903。从媒体分布 DS 903 获得通用的代码转换提示 DS 905，它将参照图 10 描述。

图 10 描述了代码转换提示 DS 1001 的结构组织，它包括参照图 11 描述的源格式规定 DS 1002 中一个实例，目标格式规定 DS1003 的一个或多个实例
 30 将参照图 12 进行描述。另外，代码转换提示 DS 1001 包括参照图 13 描述的通用的代码转换提示 DS 1004 的一种选择实例，以及参照图 15 描述的一

个选择代码转换编码复杂性 DS 1005。

图 11 根据本发明实施例，描述了与整个 A/V 内容或与特定的 A/V 段相关的源格式规定代码转换提示元数据(例如图 10 的源格式规定 DS 1002)。如图 11 所示，相关的描述符和描述方案可以包括：

- 5 • bitrate 是<int>类型并且描述源 A/V 数据流的每秒位速率。
- size_of_pictures 是<2*int>类型并且描述在 x 和 y 方向上源 A/V 格式的图像大小。
- number_of_frames_per_second 是<int>类型并且描述源内容的每秒帧数量。
- 10 • pel_aspect_ratio 是<float>类型并且描述像素宽高比。
- pel_colour_depth 是<int>类型并且描述颜色深度。
- usage_of_progressive_interlaced_format 是<1 位>大小并且描述源格式是逐行还是隔行格式。
- usage_of_frame_field_pictures 是<1 位>大小并且描述使用帧还是字段图像。
- 15 • compression method 是<int>类型并且规定用于源格式的压缩方法，可以从包括：MPEG-1、MPEG-2、MPEG-4、DV、H.263、H.261 等的列表中选择。对于每种压缩方法，可以在这里规定其他的参数。
- GOP_structure 是 I、P、B 状态的行程编码的数据字段。例如，在 MPEG-2
- 20 视频中只有 I 帧的情况下，直接变换到压缩区域的 DV 格式是可能的。

图 12 根据本发明实施例，描述了目标格式规定代码转换提示元数据，它可以与整个 A/V 内容或与特定的 A/V 段相关。如图 12 所示，相关的描述符和描述方案可以包括：

- bitrate 是<int>类型并且描述目标 A/V 数据流的每秒位速率。
- 25 • size_of_pictures 是<2*int>类型并且描述在 x 和 y 方向上目标 A/V 格式的图像大小。
- number_of_frames_per_second 是<int>类型并且描述目标内容的每秒帧数量。
- pel_aspect_ratio 是<float>类型并且描述像素宽高比。
- 30 • pel_colour_depth 是<int>类型并且描述颜色深度。
- usage_of_progressive_interlaced_format 是<1 位>大小并且描述目标格式

需要逐行还是隔行。

•usage_of_frame_field_pictures 是<1 位>大小并且描述使用帧还是字段图像。

•compression_method 是<int>类型并且规定用于目标格式的压缩方法，
5 可以从包括：MPEG-1、MPEG-2、MPEG-4、DV、H.263、H.261 等的列表
中选择。对于每种压缩方法，可以在这里规定其他的参数。

•GOP_structure 是 I、P、B 状态的可选的行程编码数据字段。借助于这个
可选参数，可以强迫一个固定的 GOP 结构。固定的 GOP 结构可以例如将
I 帧强迫在某一位置以便于视频编辑。

10 图 13 根据本发明实施例，描述了通用的代码转换提示元数据(例如，图
11 的通用代码转换提示 DS 1004)，它可以与整个 A/V 内容或与特定的 A/V
段相关。如图 13 所示，相关的描述符和描述方案可以包括：

•use_region_of_interest_DS 具有<1 位>的长度并且表示是否感兴趣的描
述方案的区域可用于代码转换提示。

15 •在使用 region_of_interest_DS 的情况下，shape_D(它可以例如是下面的
一个：boundary_box_D，MB_shape_D 或任何其它的 shape_D)与
motion_trajectory_D 一起可以用于在空间上和时间上描述感兴趣的区域。
MB_shape_D 可以使用宏块(16×16)大小的块用于目标形状描述。
Motion_trajectory_D 已经包括时间的概念，使得可以规定
20 region_of_interest_DS 的开始帧和结束帧。region_of_interest_DS 可以具有相
应的 shape_D 和相应的 motion_trajectory_D 的大小。对于代码转换应用，例
如可以使用 region_of_interest_DS，以对于感兴趣区域内的块比背景花费更
多的位(或相应地修改量化器)。对于 MPEG-4 的另一个代码转换应用可以通
过分开 MPEG-4 目标来描述感兴趣的区域，并且对于感兴趣的区域比其它的
25 MPEG-4 目标如背景花费更高的位速率和更高的帧频。region_of_interest_DS
的提取可以自动地或手工地完成。

•use_editing_effects_transcoding_hints_DS 具有<1 位>的长度并且指示是
否信息可以根据基于编辑效果的代码转换提示来获得。

30 •camera_flash 是项目的列表，这里每个项目描述摄像机闪光产生的帧数
量。因此，描述符的长度是摄像机闪光事件的数量乘以<int>。对于代码转换
应用，camera_flash 描述符是非常有用的，因为大多数视频(重新)编码器/代

码转换器使用的运动估计方法基于亮度差别，参照 Peter Kuhn “用于 MPEG-4 运动估计的算法、复杂性分析和 VLSI 结构”，Kluwer Academic Publishers，1999。在基于亮度的运动估计的情况下，两个相邻帧(一个具有闪光，一个不具有闪光)的两个宏块之间的平均绝对误差对于预测来说太高，具有摄像机闪光的帧必须作为具有高位速率成本的帧内编码。因此，表示代码转换提示描述方案(“DS”)内的摄像机闪光允许例如使用亮度校正运动估计方法或其它手段从具有适当位成本的锚帧预测具有摄像机闪光的帧。

5 •cross_fading 是项目列表，这里每个项目描述交叉衰落的开始帧和结束帧。因此，这个描述符的长度是交叉衰落事件数量的<int>的两倍。在交叉衰落期间表示代码转换提示元数据中的交叉衰落事件对于控制位速率/量化器是非常有用的。在交叉衰落期间，预测一般被限制使用，使得用于预测误差编码的位速率增加。因为在交叉衰落期间，场景通常变得模糊，通过分别调节量化器比例、位速率或速率控制参数可以限制位速率增加。

15 •black_pictures 是项目的列表，这里每个项目描述了一个黑画面序列的开始帧和结束帧。在场景之间，特别的在家庭视频中，可能产生黑色画面。根据经验，这种结果表示一系列黑画面增加了运动补偿 DCT 编码器的位速率，因为预测仅仅被有限地使用。因此，通过分别调节量化器比例、位速率或速率控制参数，这个代码转换提示描述符可以用于限制黑画面期间的位速率。

20 •fade_in 类似于 cross_fading，并且被描述为确定渐显的开始帧和结束帧的许多项目。与交叉衰落相比，渐显从黑图像开始，因此，通过分别调节量化器比例、位速率或速率控制参数，一种眼睛的屏蔽效果可以用于限制在渐显期间的位速率。

 •fade_out 类似于 fade_in，除了在场景之后的，描述一系列黑色画面。

25 •abrupt_change 通过<int>类型的单个帧数量的列表描述，表示突然的场景或拍摄变化而没有衰落出现。这些事件通过例如图 3 的非常高和非常陡的峰值来表示。这些峰值表示新的摄像或场景的开始。abrupt_change 编辑效果与衰落效果相反。当两个视频段之间的突然变化出现时，人的视觉需要几毫秒来适应和识别新的 A/V 段的细节。人眼的这种慢慢适应的效果对于视频代码转换是有益的，例如用于为在场景或摄像突然变化之后的视频段的那些

30 开始帧，减少位速率或修改量化器比例参数。

•use_motion_transcoding_hints_DS 具有<1 位>的长度并且表示运动相关的代码转换提示元数据的使用。

•number of regions 表示下面运动相关的代码转换提示元数据有效的区域数量。

- 5 •for_every_region 用<1 位>长度字段来表示该区域是矩形还是任意形状。在区域是任意形状的情况下，使用区域描述符(包括例如形状描述符和运动轨迹描述符)。在矩形区域的情况下，使用矩形区域的大小。这个区域内的运动字段用参数运动模型来描述，对于每个帧或帧序列，它由若干个参数确定。对于代码转换，源视频的实时运动的这个运动表示可以用于限制(重新)编码部分的计算复杂性运动估计的搜索区域，也可以用于快速和有效的隔行/去隔行(帧/字段)转换并且根据视频内的运动量确定 GOP(画面组)结构。
- 10 运动表示也可以有益地用于视频的大小转换。

图 14 根据本发明实施例，描述了基于段的代码转换提示元数据(例如，图 9 的基于段的代码转换提示 DS 909 和 910)，它可以用于确定描述恒定特性的 A/V 段的(重新)编码器/代码转换器。如图 14 所示，相关的描述符和描述方案可以包括：

15

•start_frame 是<int>类型并且描述 A/V 段的代码转换提示元数据开始的帧号码。

•nframe 是<int>类型并且描述 A/V 段的长度。

20 •I_frame_location 给出描述 A/V 段内 I 帧位置的几种可能。

•select_one_out_of_the_following 是<2 位>大小并且选择下面四个 I 帧位置描述方法中的一个。

•first frame 是<1 位>大小并且是缺省的 I 帧位置。这种方法描述 A/V 段，这里只有第一帧是 A/V 段的内部帧并且用作进一步预测的锚，A/V 段内所有的其它帧是 P 或 B 帧。

25

•List of frame 给出 A/V 段内内部帧的帧数量列表。这种方法允许任意地描述 A/V 段内内部帧的位置。对于这个列表中的 k 帧，这个描述符大小是<k*int>。

•first_frame_and_every_k_frames 是<int>类型，这里段内第一帧是内部的并且 k 描述了 A/V 段内 I 帧的间隔。

30

•no_I_frame 是<1 位>大小并且描述在 A/V 段内不使用 I 帧的情况，当

A/V 段的编码基于先前段的地锚(内部帧)时它是有用的。

- quantizer_scale 是<int>类型并且描述用于 A/V 段的初始量化器比例值。
- target_bitrate 是<int>类型并且描述用于 A/V 段的每秒目标位速率。
- target_min_bitrate 是<int>类型并且描述用于 A/V 段的每秒最小目标位

5 速率(可选)。

•target_max_bitrate 是<int>类型并且描述用于 A/V 段的每秒最大目标位速率(可选)。

•use_transcoding_states 是<1 位>大小并且描述代码转换提示状态是否用于 A/V 段。

10 •transcoding_state_nr 是<int>类型并且给出用于段的代码转换提示元数据状态。代码转换提示元数据状态是到代码转换提示状态表中的一项的指针。代码转换提示状态表可以具有若干个项目，这里通过代码转换提示参数可以加上或减去新的项目。下面参照图 16 描述单个代码转换提示状态的代码转换提示元数据。

15 •add_new_transcoding_state 是<1 位>大小并且描述具有相关信息的新代码转换状态是否必须加到代码转换提示表上。在 add_new_transcoding_state 发出信号“是”的情况下，给出新代码转换提示状态的参数列表。参数列表的大小由代码转换提示状态的参数数量和代码转换提示状态的数量确定。

20 •remove_transcoding_state 是<1 位>大小的标记，表示代码转换状态是否可以去除。在可以去除代码转换状态的情况下，给出去除的代码转换状态的状态数量(类型：<int>)。

•use_encoding_complexity_description 是<1 位>大小并且发出是否必须使用如图 15 规定的更详细的编码复杂性描述方案的信号。

25 图 15 根据本发明实施例，描述了编码复杂性代码转换提示元数据，它可以与整个 A/V 内容或与特定的 A/V 段相关。编码复杂性元数据可以用于速率控制并且确定量化器和位速率设置。

•use_feature_points 是<1 位>大小并且表示基于特征点的复杂性估计数据的使用。

•select_feature_point_method 是<2 位>大小并且选择特征点方法。

30 •number_of_new_feature_points 每帧描述如图 3 说明的每帧新特征点数量的列表，并且它们是<nframes*int>大小。这个度量表示每帧的新内容量。

•feature_point_metrics 描述了一个段内每帧新特征点的度量列表。该度量被表示为具有下面意义的<int>值的有序列表：平均值、最大值、最小值、变量、每帧新特征点数量的标准偏差。

•use_equation_description 是到每帧编码复杂性的基于方程式描述的<int>

5 指针。

•use_motion_description 是<1 位>大小并且表示基于运动的复杂性描述的使用。

•select_motion_method 是<4 位>大小并且选择运动描述方法。

10 •param_k_motion 是<nframes*k*int>大小并且描述用于全局参数运动模型型的每单个帧的 k 个参数。

•motion_metrics 描述了度量列表，用于基于整个段的运动向量的大小。度量被表示为具有下面意义的<int>值的有序列表：平均值、最大值、最小值、变量、宏块运动向量的标准偏差(stddev)。

15 •block_motion_field 描述了 m*m 块大小的运动字段的每个向量并且是<nframes*int*size_x*size_y/(m*m)>的大小。

•use_texture_edge_metrics 是当结构或边缘度量被使用并且它是<1 位>大小时设置的标记。

•select_texture_edge_metrics 是<4 位>大小并且它确定来自下面的哪个结构度量被使用。

20 •DCT_block_energy 是一个块的所有 DCT 系数的总和并且被定义用于帧内每个块。它是<size_y*size_x*nframes*int/64>的大小。

•DCT_block_activity 被定义为一块的所有 DCT 系数的总和，但没有 DC 系数。它被定义用于帧内的每个块并且是<size_y*size_x*nframes*int/64>的大小。

25 •DCT_energy_metric 描述了基于每块的各个 DCT 能量的整个段的度量列表。该度量被表示为具有下面意义的<int>值的有序列表：平均值、最大值、最小值、变量、所有各个 DCT 能量度量的标准偏差。该描述符的大小是<6*int>。这个描述符的另一种实现是描述用于视频段的每单个帧的 DCT 能量度量。

30 •DCT_activity_metric 描述了基于每块的各个 DCT 活动性的整个段的度量列表。该度量被表示为具有下面意义的<int>值的有序列表：平均值、最大

值、最小值、变量、所有各个 DCT 活动性度量的标准偏差。该描述符的大小是<6*int>。这个描述符的另一种实现是描述用于视频段的每单个帧的 DCT 活动性度量。

图 16 根据本发明实施例，描述了代码转换提示状态元数据，它可以与整个视听内容或与特定的 A/V 段相关。相关的描述符和描述方案可以包括：

•M 是<int>类型并且描述 I 帧/P 帧距离。

•bitrate_fraction_for_I 是<float>类型并且描述可用于 I 帧的 A/V 段规定的位速率的分数。

•bitrate_fraction_for_P 是<float>类型并且描述可用于 P 帧的 A/V 段规定的位速率的分数。用于 B 帧的位速率分数是到 100% 的剩余百分比。

•quantizer_scale_ratio_I_P 是<float>类型并且表示 I 和 P 帧之间的量化器比例(如为这个段规定的)关系。

•quantizer_scale_ratio_I_B 是<float>类型并且表示 I 和 B 帧之间的量化器比例(如为这个段规定的)关系。应该注意位速率描述符 (bitrate_fraction_for_I<bitrate_fraction_for_P)， quantizer_scale_ratio 描述符 (quantizer_scale_ratio_I_P， quantizer_scale_ratio_I_B)或下面的速率控制参数可以是强制的。

•X_I， X_P， X_B 是 frame_vbv_complexities， 每个是<int>类型并且在基于帧的压缩目标格式的情况下规定(参照图 12)。根据源内容特征和目标格式规定， 这些和下面的虚拟缓存器检验器(“VBV”)复杂性调节是可选的并且可以用于修改速率控制方案。

•X_I top， X_P top， X_B top 是用于顶部字段的 field_vbv_complexities， 每个是<int>类型并且在基于字段的压缩目标格式的情况下规定(参照图 12)。

这样从前面的描述可以有效地达到本发明的目的， 因为在实现上面的方法和结构中可以有的一些变化但不会背离本发明的精神和范围， 上面的描述和附图中包含的内容只用于说明的目的， 而没有限制的意思。

还应该理解所附权利要求书涵盖了这里描述的本发明的一般和特定的特征以及本发明范围内的所有描述。

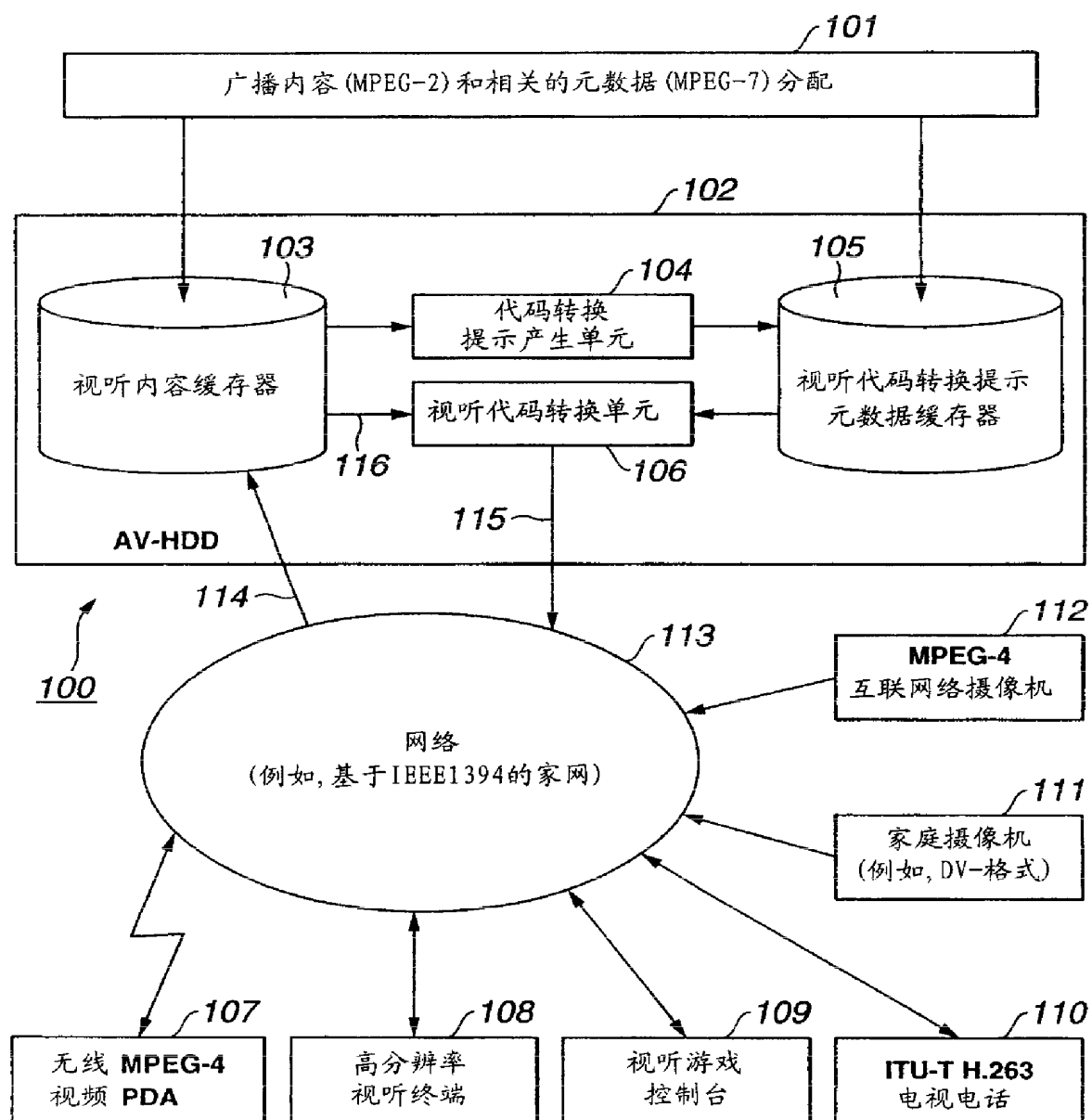


图 1

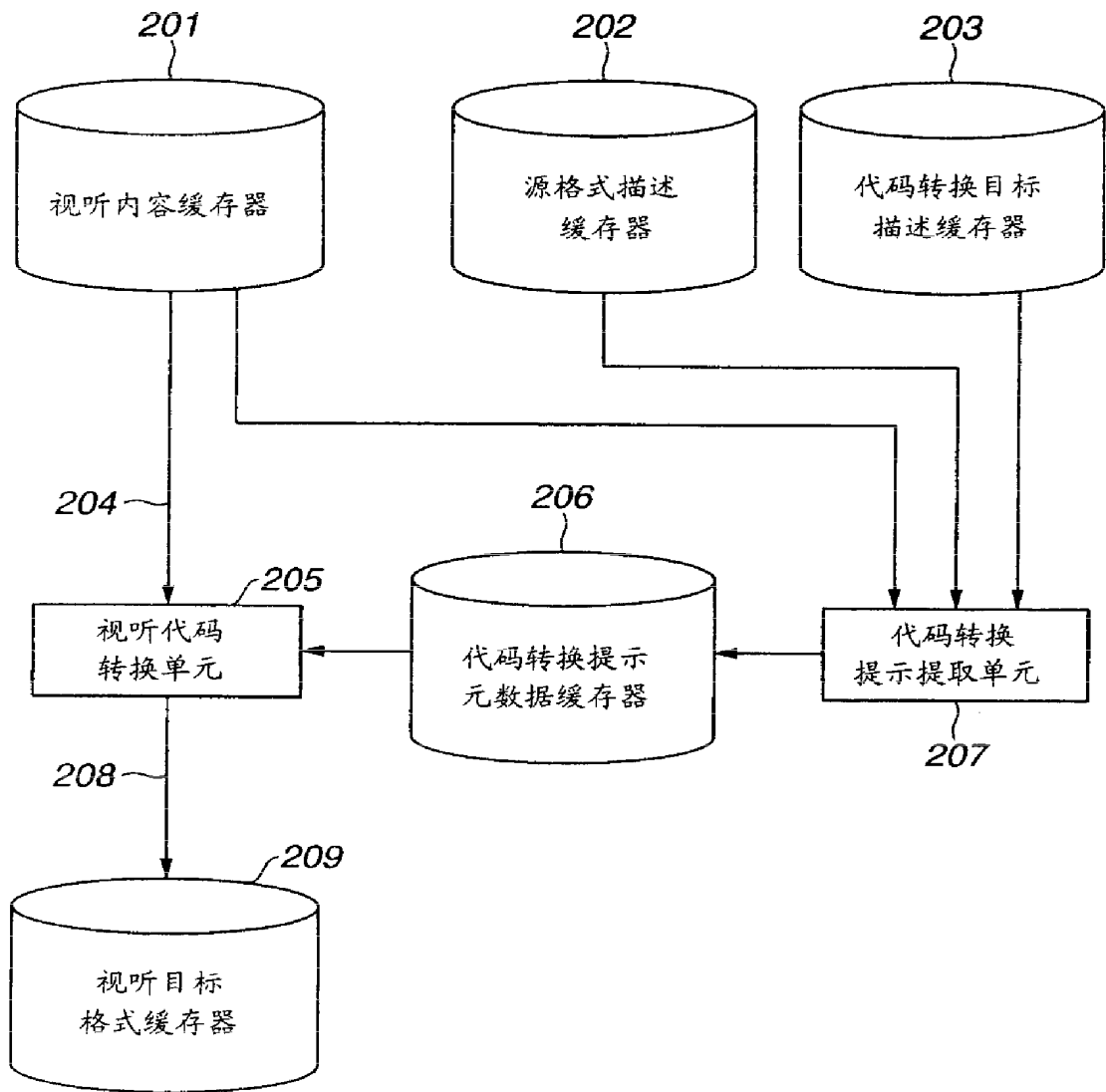


图 2

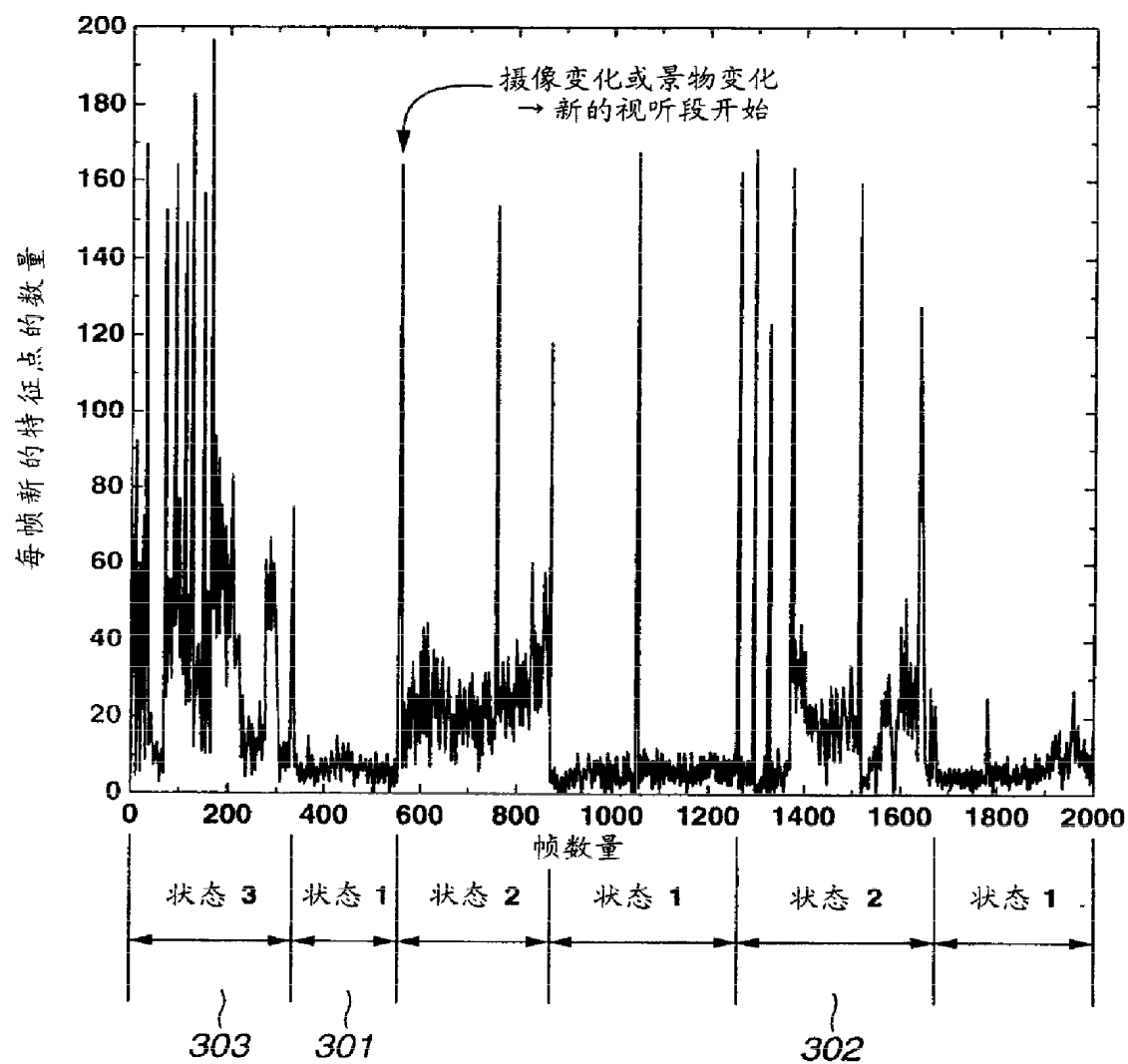


图 3

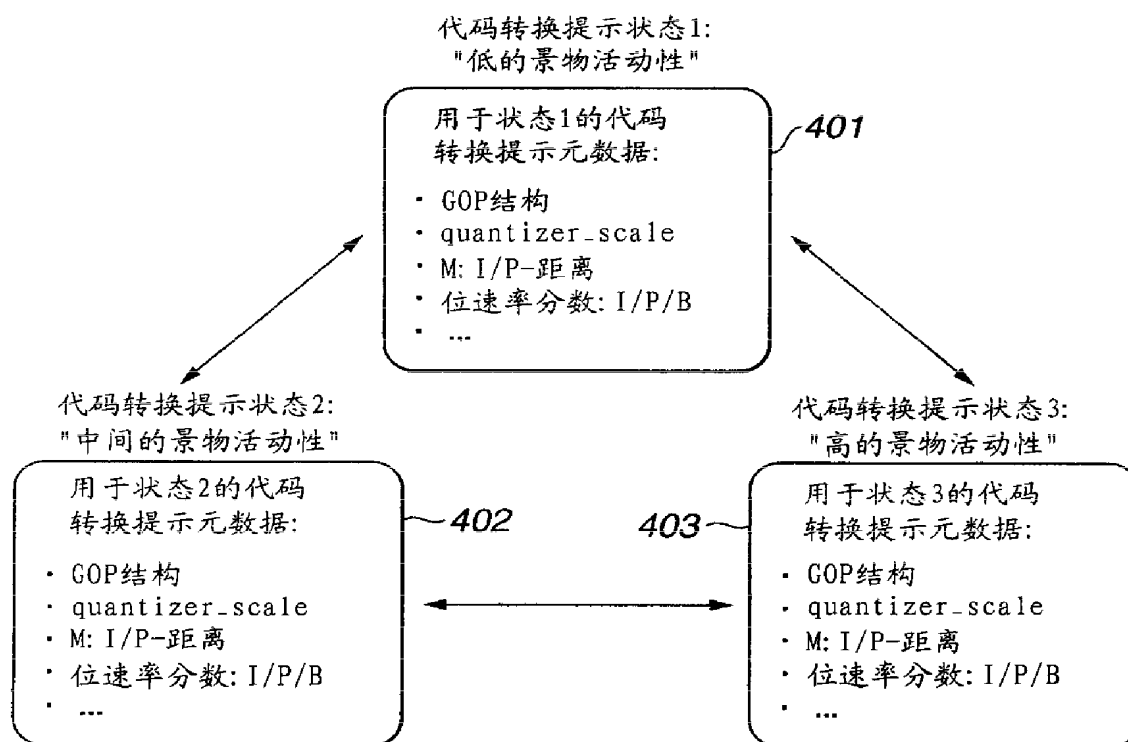


图 4

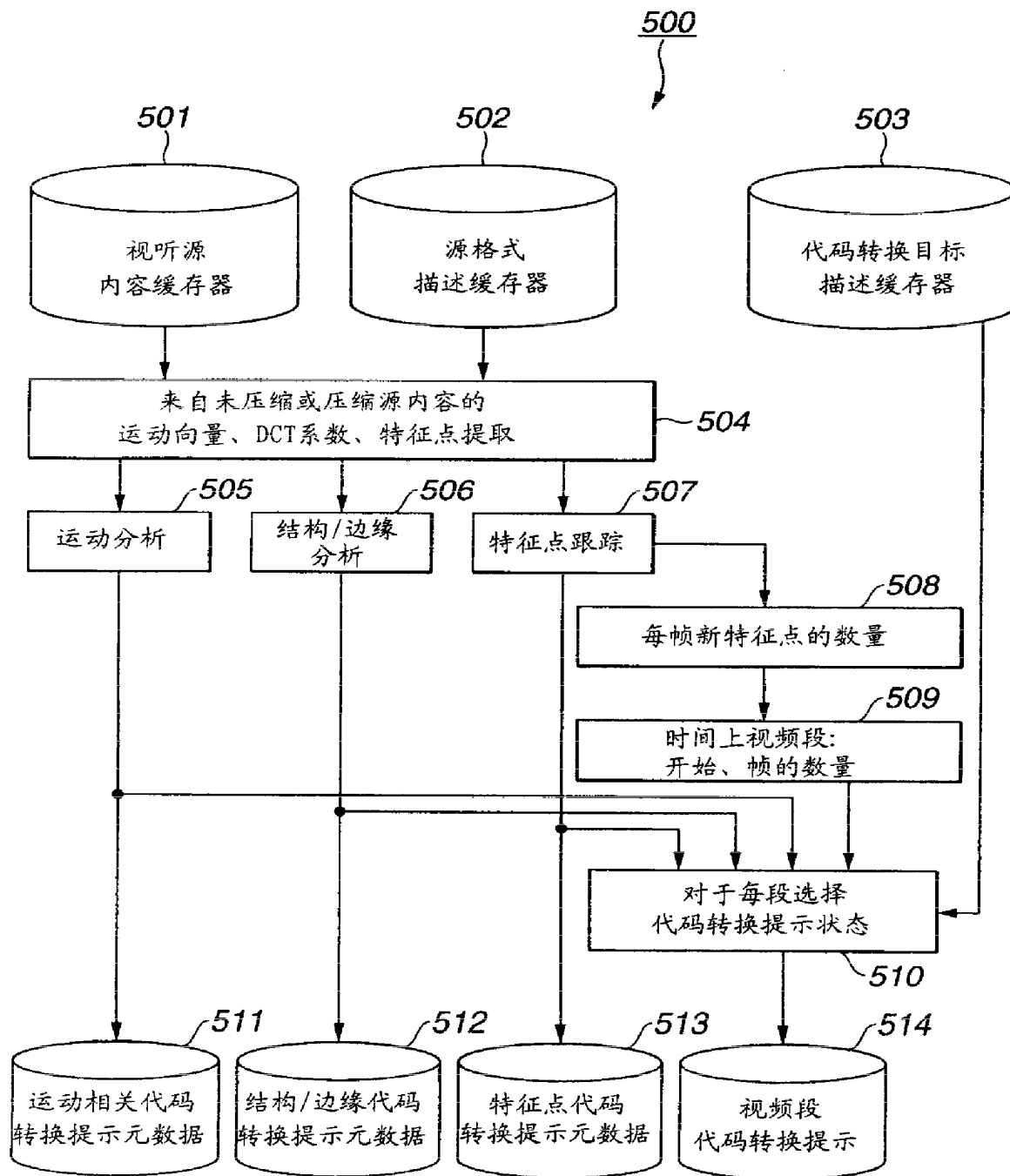


图 5

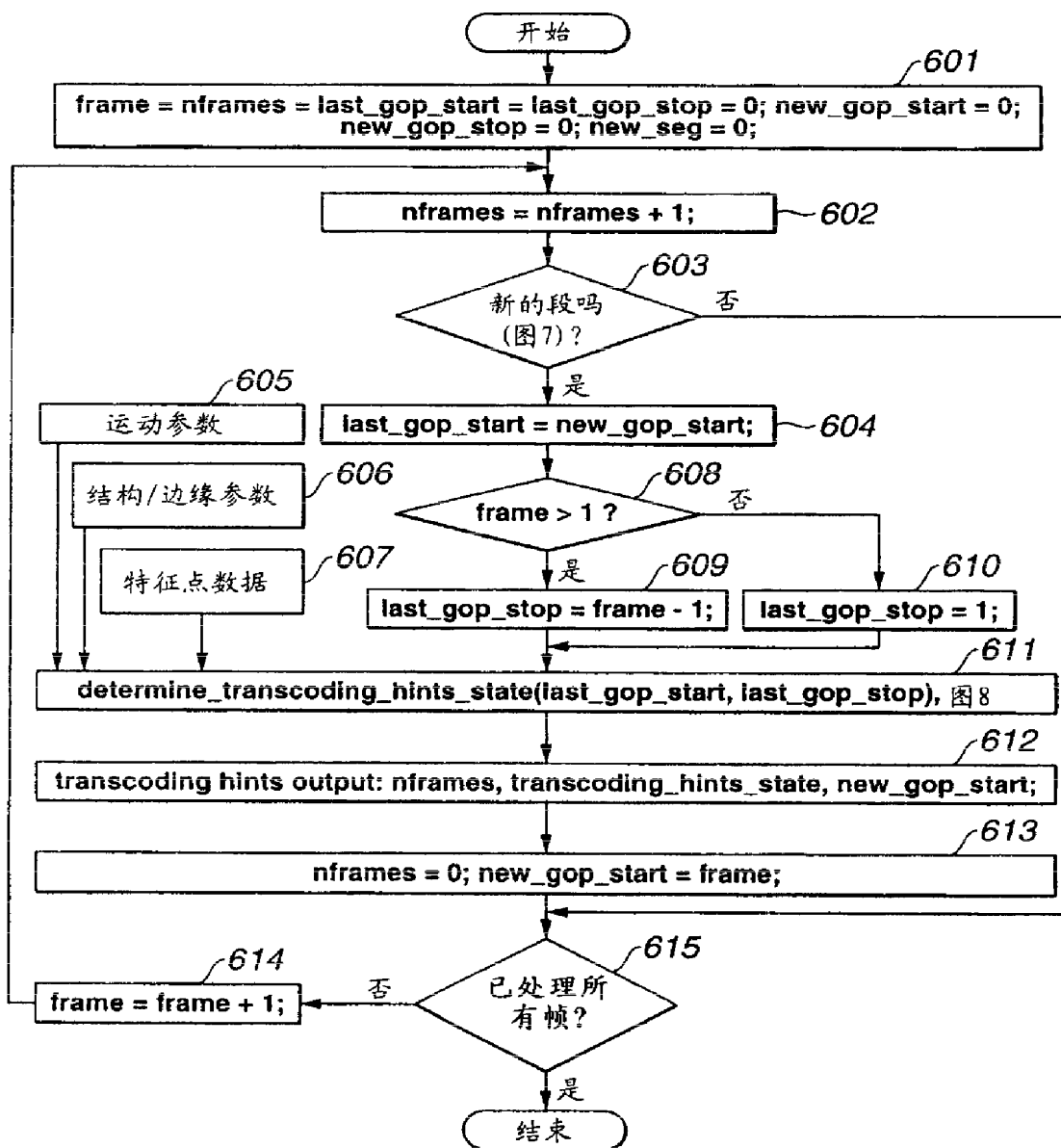


图 6

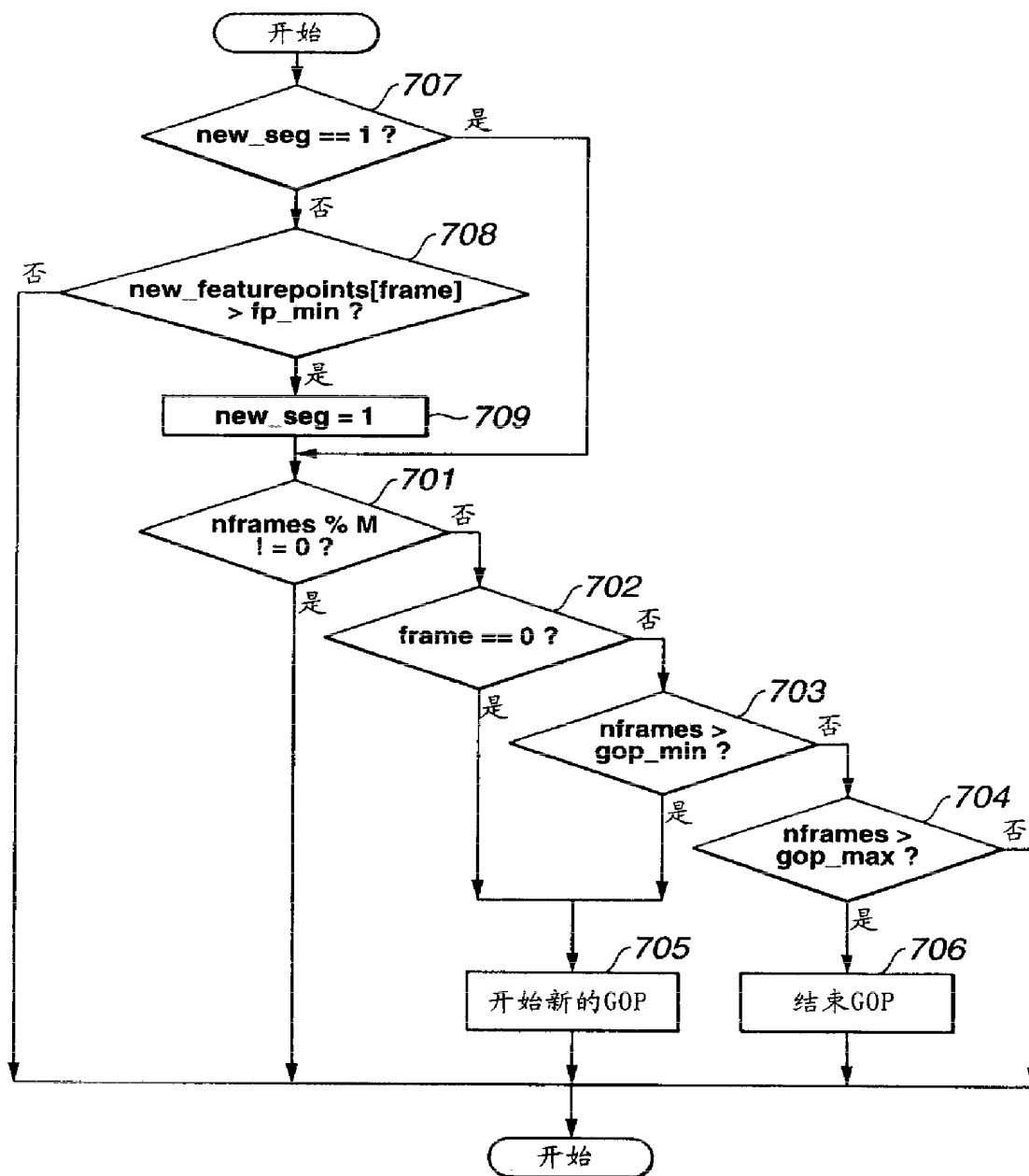


图 7

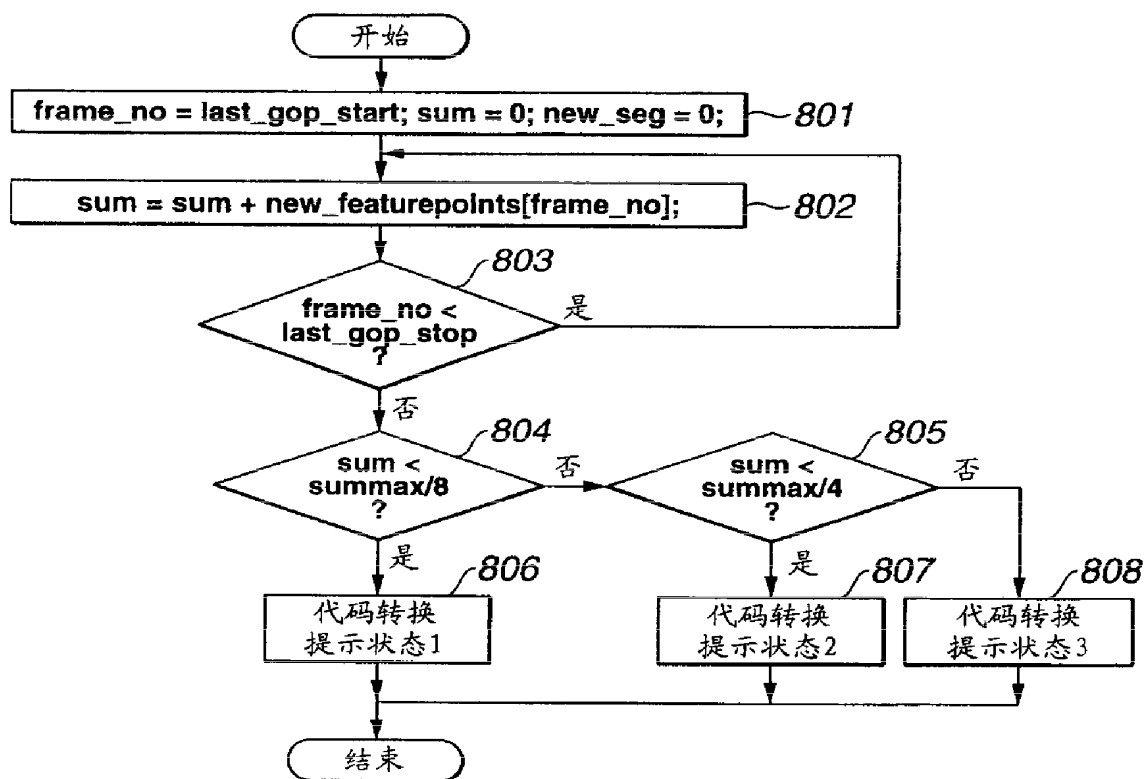


表1: 代码转换提示状态表(例子)

状态	M	bitrate_fraction_for_l	bitrate_fraction_for_p
1	5	0.8	0.15
2	4	0.85	0.1
3	3	0.9	0.05

图 8

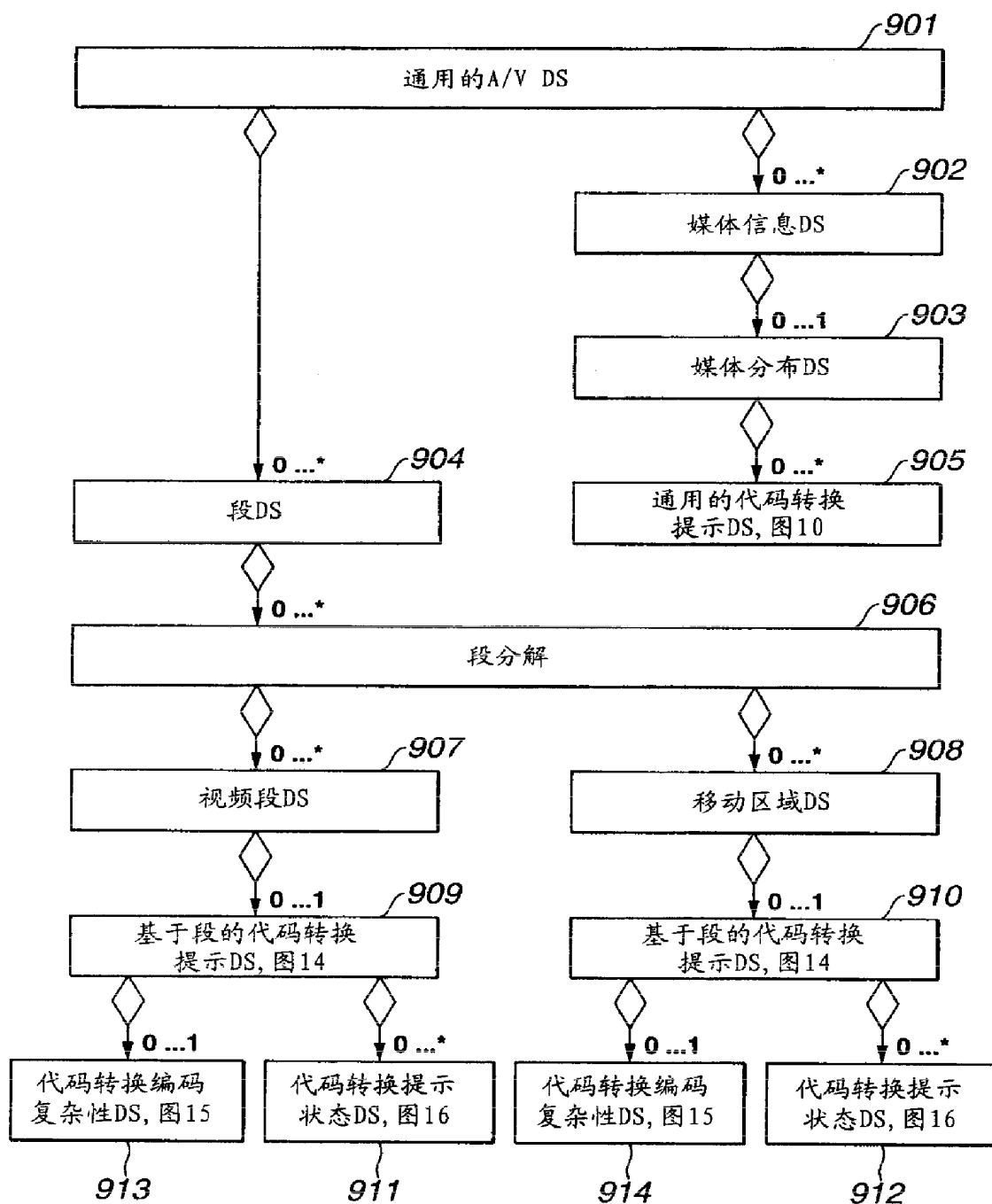


图 9

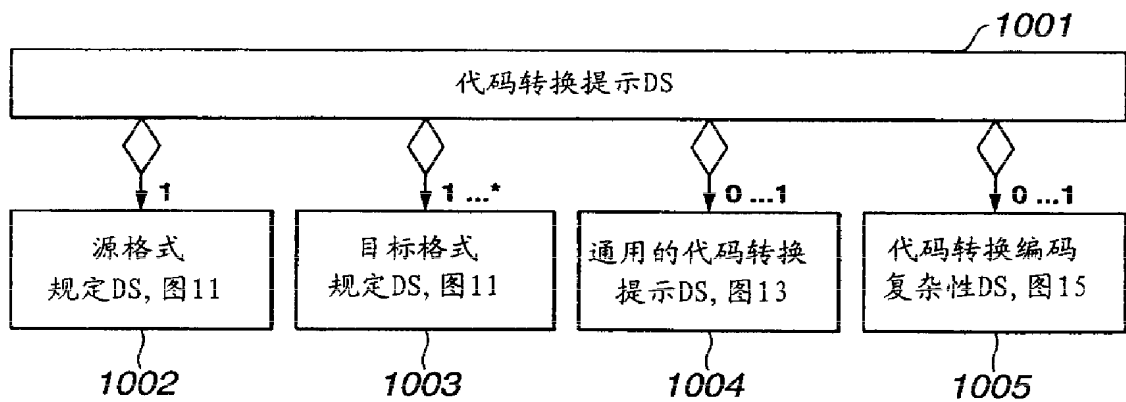


图 10

1:	bitrate <int>
2:	size_of_pictures <2*int>
3:	number_of_frames_per_second <int>
4:	pel_aspect_ratio <float>
5:	pel_colour_depth <int>
6:	usage_of_progressive_interlaced_format <1 bit>
7:	usage_of_frame_field_pictures <1 bit>
8:	compression method <int>
9:	one out of list {MPEG-1, MPEG-2, MPEG-4, DV, H.263, H.261, }
10:	{ further parameters for compression method }
11:	GOP_structure (Runlength coding)

图 11

```

1: bitrate <int>
2: size_of_pictures <2*int>
3: number_of_frames_per_second <int>
4: pel_aspect_ratio <float>
5: pel_colour_depth <int>
6: usage_of_progressive_interlaced_format <1 bit>
7: usage_of_frame_field_pictures <1bit>
8: compression_method <int>
9:     one_out_of_list {MPEG-1, MPEG-2, MPEG-4, DV, H.263, H.261, .... }
10:    {用于压缩方法的其他参数 }
11:    GOP_structure (行程编码 )

```

图 12

```

1: use_region_of_interest_DS: <1bit>
2:     region_of_interest_DS:
3:         shape_D: select one or {boundary_box_D, MB_shape_D, shape_D}
4:         motion_trajectory_D
5:
6: use_editing_effects_transcoding_hints_DS: <1bit>
7:     camera_flash {frame1, frame2, .... framek} <k*int>
8:     cross_fading {(start_frame, end_frame), ... } <k*(<int>, <int>)>
9:     black_pictures {(start_frame, end_frame), ... } <k*(<int>, <int>)>
10:    fade_in {(start_frame, end_frame), ... } <k*(<int>, <int>)>
11:    fade_out {(start_frame, end_frame), ... } <k*(<int>, <int>)>
12:    abrupt_change {frame1, frame2, .... framek} <k*int>
13:
14: use_motion_transcoding_hints_DS: <1 bit>
15:     number_of_regions: <int>
16:     for_every_region:
17:         is_region_rectangular_shaped (y/n): <1bit>
18:         if_arbitrarily_shaped: 使用区域D用于这个区域
19:         描述用于这个区域的参数目标运动

```

图 13

```

1: start_frame <int>
2: nframes <int>
3: I_frame_location:
4:     select_one_out_of_the_following: <2 bit>
5:         first frame (default)
6:         list of frames {frame1, frame2, . . . . , framek} <k*int>
7:         first_frame_and_every_k_frames <int>
8:         no_I_frame
9: quantizer_scale <int>
10: target_bitrate <int>
11: target_min_bitrate <int>
12: target_max_bitrate <int>
13: use_transcoding_states (y/n) <1 bit>
14: transcoding_state_nr <int>
15: add_new_transcoding_state (y/n) <1bit>
16:     if yes: {list of parameters}
17: remove_transcoding_state (y/n) <1bit>
18:     if yes: state_nr <int>
19: use_encoding_complexity_description (y/n) <1 bit>
20:     if yes: encoding_complexity_description_scheme

```

图 14


```

1: use_feature_points (y/n) <1bit>
2:     select_feature_point_method <2 bits>
3:         number_of_new_feature_points <nframes * int>
4:         feature_point_metrics {mean, max, min, var, stddev} <5* int>
5: use_equation_description (y/n) <1bit>
6: use_motion_description (y/n) <1bit>
7:     select_motion_method <4 bits>
8:         param_k_motion <nframes * k * int>
9:         motion_metrics {min, max, sum, var, stddev} <5*int>
10:        block_motion_field <nframes*int*size_x*size_y / (m*m) >
11: use_texture_edge_metrics (y/n) <1bit>
12:     select_texture_edge_metrics <4 bits>
13:         DCT_block_energy <size_y*size_x*nframes*int/64>
14:         DCT_block_activity <size_y*size_x*nframes*int/64>
15:         DCT_energy_metric {mean, min, max, sum, var, stddev} <6*int>
16:         DCT_activity_metric {mean, min, max, sum, var, stddev} <6*int>

```

图 15

```

1: M: I/P distance <int>
2: bitrate_fraction_for_I <float>
3: bitrate_fraction_for_P <float> /* bitrate_fraction of B is rest to 100 % )
4: quantizer_scale_ratio_I_P <float>
5: quantizer_scale_ratio_I_B <float>
6: if_frame: /* see target format transcoding hints */
7:     X_I, X_P, X_B <3*int> /* frame_vbv_complexities */
8: if_field:
9:     X_I_top, X_P_top, X_B_top <3*int> /* field_top_vbv_complexities */
10:    X_I_bot, X_P_bot, X_B_bot <3*int> /* field_bottom_vbv_complexities */

```

图 16